

# PYTHON STANDARD LIBRARY

**Webbrowser, configparser, namedtuple**  
**Elliot Garbus**  
**5/22/2019**

# AGENDA

- About Me
- Session Objective
- Learning Resources
- Demonstration
- Webbrowser
- ConfigParser
- NamedTuple



# ABOUT ME

- Management Consultant and Python Hobbyist & Enthusiast
- As a professional developer:
  - Primarily ASM, C and C++ as a professional dev (Embedded/Telecom)
  - Lots of ‘on the metal’ programming and performance optimization.
  - Worked closely with C/C++ compiler teams on low-level optimization, and on performance tools
  - ...all a very long time ago



# SESSION OBJECTIVES

Create awareness of a few standard library features

- Webbrowser, configparser, Collections.namedtuple

An introduction not an in depth review

Showcase how I have used these libraries in a project

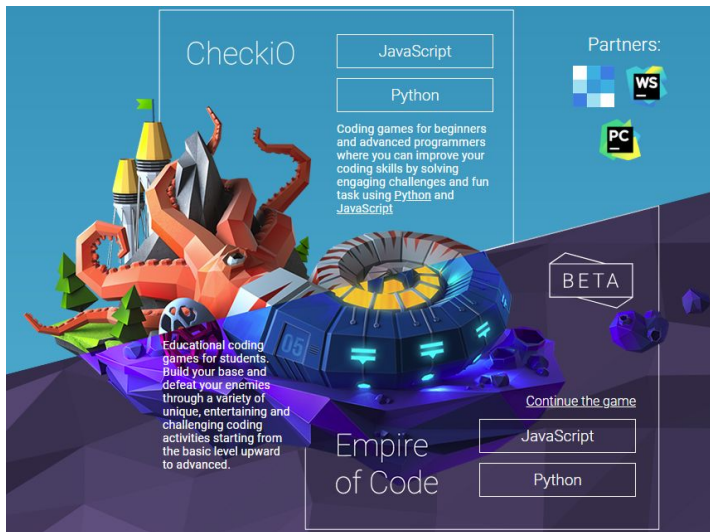
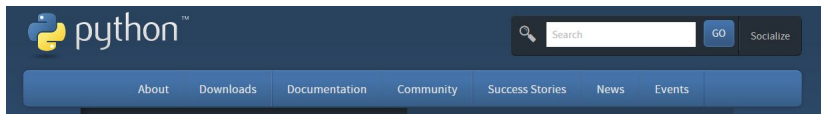


# LEARNING RESOURCES

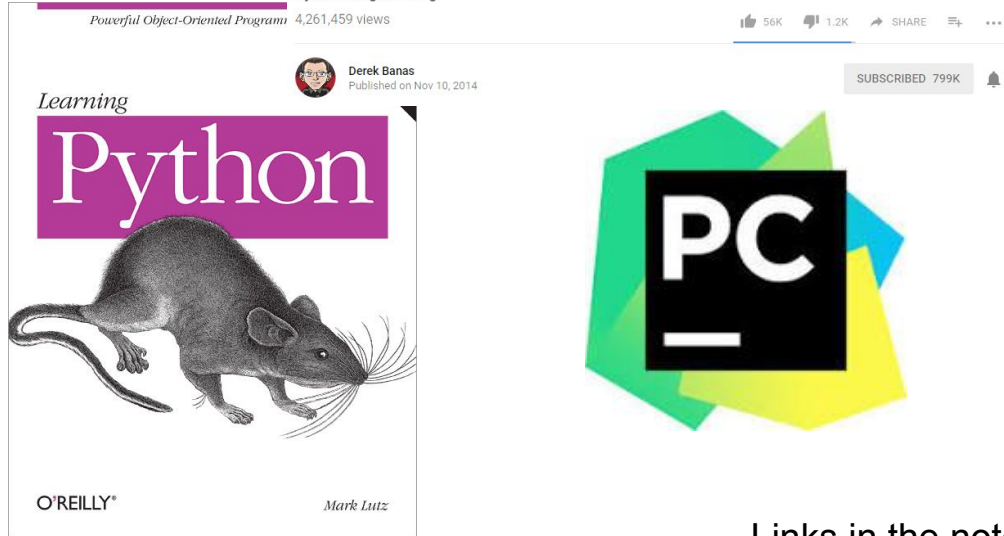
PyMOTW-3

## Python 3 Module of the Week

PyMOTW-3 is a series of articles written by [Doug Hellmann](#) to demonstrate how to use the modules of the [Python](#) 3 standard library. It is based on the original [PyMOTW](#) series, which covered Python 2.7. See [About Python Module of the Week](#) for details including the version of Python and tools used.



Python Programming



# LINKS

<https://pymotw.com/3/>

<https://www.python.org/>

<https://www.youtube.com/watch?v=N4mEzFDjqtA>

[https://www.amazon.com/\\_/dp/1449355730?tag=oreilly20-20](https://www.amazon.com/_/dp/1449355730?tag=oreilly20-20)

<https://checkio.org/>

<https://www.jetbrains.com/pycharm/>

WEBBROWSER

# WEBBROWSER

- High-level interface to display Web-pages.
- Under most circumstances, simply calling `open()` will do the right thing
- My use: Provide access to users manual & intro videos

```
#import webbrowser
```

```
webbrowser.open('www.youtube.com')
```





CONFIGPARSER

# CONFIGPARSER

- Implements a basic configuration language
  - Dictionary like interface
  - Lots of options for extended capabilities, legacy formats
- File similar to Microsoft Windows INI files
- My use:
  - Preserve window size and position across sessions
  - Save connection interface, and connect automatically
  - Save user preferences for file locations
- Alternatives:
  - Could use JSON to implement similar capabilities
  - A number of libs on PyPI ConfigObj...

# CONFIGPARSER:EXAMPLE FILE

[MIDI]

input = 2- UM-ONE 0  
output = 2- UM-ONE 1  
channel = 1

[Enzo]

lib\_dir = .  
default\_dir = .  
man\_url = [https://www.meris.us/wp-content/uploads/2018/06/Meris\\_Enzo\\_Manual\\_v1c.pdf](https://www.meris.us/wp-content/uploads/2018/06/Meris_Enzo_Manual_v1c.pdf)  
intro\_vid\_url = <https://youtu.be/AEB-HyNV6LU>

[Window]

width = 1275  
height = 660  
top = 68  
left = 181

# CONFIGPARSER: API OVERVIEW

`read(filename, encoding=None)`

- Read a file, or list of files, returns list of filenames or an empty list.

`get(section, option)`

`add_section(section)`

`set(section, option, value)`

`write()`



# CONFIGPARSER: EXAMPLES

- Simple Example
- My use case



# NAMEDTUPLES

# NAMEDTUPLES



## Factory Function for Tuples with Named Fields

- Assign meaning to each position in a tuple
- More readable, self-documenting code
- Access fields by name instead of index
- My Use:
  - Access fields by name, easier to maintain, errors less likely
  - Simplified print based debug
  - Used to catch exceptions from short data messages

One of many useful ‘Containers’ in the Collections Library

# NAMEDTUPLES:SIMPLE EXAMPLE

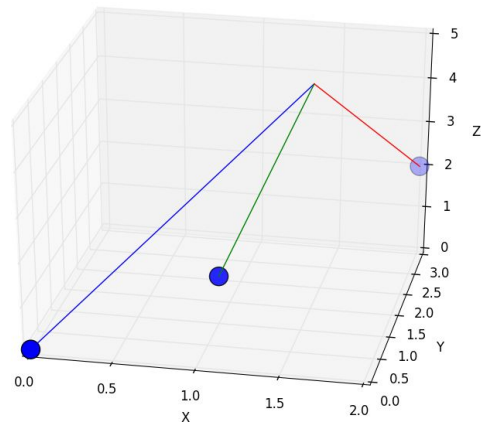
Imagine we have a list of points with the values (x,y,z)

Tuple access, by index

- `point[0]` is x; `point[1]` is y; `point[2]` is z

NamedTuple access:

- `point.x`, `point.y`, `point.z` or by index





# NAMEDTUPLES

- Simple Example
- My Use Case



# SUMMARY

## Webbrowser

- Open a browser and display a page

## Configparser

- \*.ini files for initialization and persistence

## Collections.namedtuple

- Assign meaning to each position in a tuple